MSTWG Multistatic Tracker Evaluation Using Simulated Scenario Data Sets

Doug Grimmett
SPAWAR Systems Center San Diego (SSC-SD)
53560 Hull Street, San Diego, CA, 92152-5001, U.S.A.
grimmett@spawar.navy.mil

Stefano Coraluppi
NATO Undersea Research Centre
Viale S. Bartolomeo 400, 19126 La Spezia, Italy
E-mail: coraluppi@nurc.nato.int

Brian R. La Cour
Applied Research Laboratories
The University of Texas at Austin, Austin, TX
E-mail: blacour@arlut.utexas.edu

Christian G. Hempel
Naval Undersea Warfare Center
1176 Howell St., Newport, RI 02841 USA
Email: hempelcg@npt.nuwc.navy.mil

Thomas Lang
General Dynamics Canada Limited
3785 Richmond Road, Ottawa, Ontario, Canada
E-mail: tom.lang@gdcanada.com

Pascal A.M. de Theije
TNO Defence, Security and Safety
P.O. Box 96864, 2509 JG The Hague, The Netherlands
pascal.detheije@tno.nl

Peter Willett
ECE Dept., University of Connecticut
Storrs, CT 06269
Email: peter.willett@uconn.edu

Abstract *- The Multistatic Tracking Working Group (MSTWG) was formed in 2005 by an international group of researchers interested in developing and improving tracking capabilities when applied to multistatic sonar and radar problems. The MSTWG developed several simulated multistatic sonar scenario data sets for use in tracker evaluation by the group's participants. A common set of performance metrics was also agreed, to enable tracker algorithm comparison and evaluation. Previous conference special sessions of the MSTWG have reported individual algorithm performance on these data sets. In this paper, the various results are consolidated in order to make a first attempt at performance cross-comparisons. The data sets are reviewed and performance results are presented. Issues with various performance metrics are explained.*

Keywords: Multistatic Sonar, Multistatic Radar, Multi-Sensor Fusion , Tracking, MSTWG

1 Introduction

Distributed multistatic active sonar networks have the potential to increase ASW performance against small, quiet threat submarines in the harsh clutter-saturated littoral and the deeper open ocean. This improved performance comes through the expanded geometric diversity achieved with multiple sources and receivers, and results in increased probability of detection, area coverage, target tracking, classification, and localization through cross-fixing [1].

However, with the increased number of sensors in a multistatic network come corresponding increases in the data rate, processing, communications requirements, and operator loading. Without an effective fusion of the multistatic data, the benefits of such systems will be unrealizable. Effective, robust, and automated multi-sensor data fusion and tracking algorithms become an essential part of such systems.

The Multistatic Tracking Working Group (MSTWG) was organized in 2005, with overall objectives:

- Fostering the exchange of scientific and technical ideas, problems, and solutions related to multistatic tracking for sonar and radar.
- Collaborative analysis and evaluation of multistatic tracking algorithms, applied to common data sets using a common set of metrics. It is expected that each tracking approach will exhibit strengths and weaknesses in a scenario-dependent and metric-dependent manner. The goal is to capture these effects and better understand algorithm differences and their applicability.

The working group has met once to twice a year since 2005, in the following locations: The Hague, Netherlands; Bonn, Germany; Florence, Italy; Aberdeen, Scotland; La Spezia, Italy; and Cologne, Germany. The meetings in Florence, Aberdeen, and Cologne were in conjunction with special sessions on multistatic tracking held at the FUSION'06, OCEANS'07, and FUSION'08 conferences, respectively. The proceedings for these conferences

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE **JUL 2008**	2. REPORT TYPE	3. DATES COVERED **00-00-2008 to 00-00-2008**
4. TITLE AND SUBTITLE **MSTWG Multistatic Tracker Evaluation Using Simulated Scenario Data Sets**		5a. CONTRACT NUMBER 5b. GRANT NUMBER 5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)		5d. PROJECT NUMBER 5e. TASK NUMBER 5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **SPAWAR Systems Center San Diego,(SSC-SD),53560 Hull Street,San Diego,CA,92152-5001**		8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)		10. SPONSOR/MONITOR'S ACRONYM(S) 11. SPONSOR/MONITOR'S REPORT NUMBER(S)
12. DISTRIBUTION/AVAILABILITY STATEMENT **Approved for public release; distribution unlimited**		
13. SUPPLEMENTARY NOTES **11th International Conference on Information Fusion, June 30 ? July 3, 2008, Cologne, Germany.**		
14. ABSTRACT **see report**		
15. SUBJECT TERMS		

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**8**	

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std Z39-18

contain a complete archive of the published MSTWG-related papers. This paper extracts results from these papers for cross-evaluation. An overview of MSTWG, including an initial description of the commonly agreed performance metrics, is available [2]. In 2008, the MSTWG was formalized as a working group under the auspices of the International Society of Information Fusion (ISIF).

2 Tracker Evaluation Matrices

The MSTWG evaluation matrices contain quantitative performance measures of multiple applications of seven different algorithms, applied to three different simulated scenarios. There are up to nine different performance metrics, and different applications of algorithms use different input parameters. The matrices are currently sparsely populated, as researchers have been endeavoring to design their algorithms and exercise them on the MSTWG data sets. Some algorithms are more mature; others are in the design stages. As results become available, they are entered in the performance matrices. There are now enough data to begin algorithm comparisons. An objective of this effort is to initiate more in-depth discussions between MSTWG members to understand algorithm differences which explain the results. The focus is not on identifying “winners” and “losers” as much as it is to understand why one algorithm performs better for a particular metric, on a particular scenario.

2.1 Simulation Scenarios

There are three multistatic scenarios with simulated data which have been made available to the MSTWG for tracker evaluation. The characteristics of these data sets are summarized below. More details for each scenario are given in the references, as well as in later sections of this paper.

- The NURC-provided data set [3]. This data set contains simulated contact data, with 4 sonar nodes (1 monostatic, 3 bistatic). A single target is modeled using the sonar equation, and random false contacts are produced according to a Rayleigh distribution. Results for this scenario are found in section 3 of this paper.

- The ARL/UT-provided data set [4]. This data set uses acoustic data collected from an actual sea trial (DEMUS’04). Two targets (one slow moving, the other fast) are modeled and injected into the hydrophone data, which is then processed into sonar detection contacts. There are two bistatic nodes (source-receiver pairs). Although some analysis has been made of this scenario, there are not yet any reported performance metrics suitable for cross-evaluation.

- The TNO-provided data set [5]. This data set is generated at the hydrophone time series level, and then processed into sonar detection contacts. There are three targets modeled; one with a maneuvering trajectory, and two representing fixed features. There are four sonar nodes; two monostatic and two bistatic. Results for this scenario are found in section 4 of this paper.

2.2 MSTWG Algorithms

The following algorithms have been applied to one or more of the MSTWG simulated scenarios, with varying levels of analysis performed. They are designated by their originating organization. More detailed descriptions of the individual algorithms and their performance results are available in the references.

- ARL/UT [6]: A Bayesian tracking method which represents the posterior probability distribution as an ensemble of sample points.

- GDCAN [7]: A two-level distributed Multi-Hypothesis Tracker (MHT), architecture. Data from common receivers are associated first, followed by fusion across receivers. The implementation is done in Cartesian coordinates using a linear Kalman Filter.

- NURC [8-9]: The primary NURC tracker is a distributed MHT design, which was used in the analyses of ARL/UT and TNO scenarios (though full sets of metrics were not obtained). For the NURC scenario, a simpler baseline tracker was applied. The baseline tracker uses an extended Kalman filter, logic based track management, and a centralized architecture.

- NUWC [10]: A Probabilistic Multiple Hypothesis Tracking (PMHT), adapted for multistatic use, and utilizing amplitude information.

- SSC-SD [11-12]: The “SPECSweb” tracker uses specular echoes as cues to selectively retrieve only a small (but relevant) subset of the sensor data for ingestion into the algorithm. The approach uses two thresholds, reverse-time tracking, and is implemented with a linear Kalman Filter.

- TNO [13-14]: A logic-based centralized Probabilistic Data Association (PDA) algorithm using an extended Kalman Filter and adapted for multistatic use.

- UConn [15,16]: Two different tracking approaches are applied: 1) a maximum likelihood probabilistic data association (ML-PDA) algorithm, adapted to work in sequential, rather than batch mode, and 2) a Gaussian mixture cardinalized probability hypothesis density tracker (GM-CPHD).

Other MSTWG participants (APL/UW, DRDC Atlantic, FGAN, Metron) have so far been unable to apply algorithms to the data sets, but have contributed to the group through valuable information exchange. Table 1 shows an overview of the MSTWG activity to date.

Table 1. Overview of MSTWG algorithm application (✓ - results with full metrics; a – scenario analyzed, not all metrics calculated)

ORG // Scenario:	NURC	ARL/UT	TNO
APL/UT (US)	a	a	a
ARL/UW(US)			
DRDC (CA)			
FGAN (GE)			
GDCAN (CA)	✓	✓	✓
Metron (US)			
NURC (NATO)	✓	a	a
NUWC (US)	a		
SSC-SD (US)	✓		
TNO (NL)	✓		✓
UConn (US)	✓	✓	✓

2.3 Algorithm Run

Each algorithm may be run multiple times on a single scenario. The algorithms may have been run with different input parameters, such as for data input thresholding or track initialization, etc. In the results that follow, these different tracker runs (for each scenario) are designated by a numeric index (1, 2, 3, etc.) following the algorithm identification.

2.4 Performance Metrics

Figure 1 shows a selection of past analysis products generated by different MSTWG researchers on the three data sets. The first, second, and third rows correspond to the ARL/UT, NURC, and TNO simulated scenarios, respectively. Though a picture may be worth a thousand words, when attempting comparison amongst different algorithms they are insufficient to completely characterize and evaluate the various results. Therefore, the MSTWG has developed and agreed upon a number of quantitative performance metrics, which are defined in detail in [2], and which will also be discussed in this paper. The goal is to capture the main elements of tracker performance with a small set of metrics that relate to operational effectiveness. The list of performance metrics is summarized below:

- Tracker Input Metrics (based on contacts)
  - PD: Contact Probability of Detection
  - FAR: False Alarm (Contact) Rate
  - LE: Contact Localization Error
- Tracker Output Metrics (based on tracks)
  - T-PD: Tracker PD (holding fraction)
  - T-FAR: False Track Rate
  - T-LE: Track Localization Error
  - TF: Tracker Fragmentation
  - L: Latency (time lag)
  - ER: Execution Rate (v. real time)

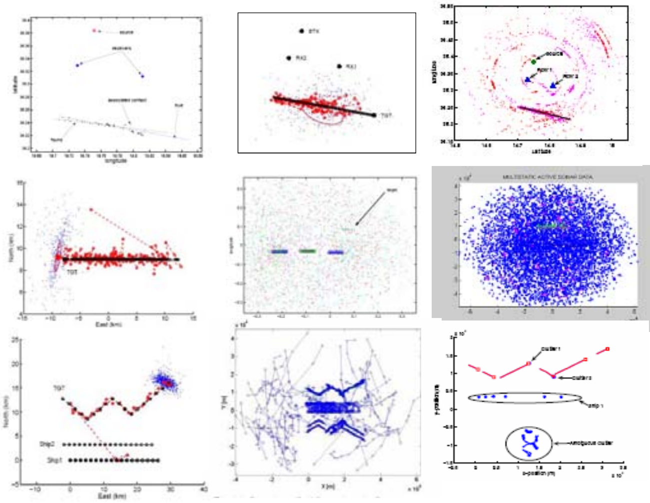

Figure. 1. Without the use of quantitative metrics, analysis products (such as these shown for various MSTWG scenarios and algorithms) are difficult to compare.

Performance comparisons between tracking algorithms require the use of common output metrics, such as these. In addition, for valid comparisons, there must be equivalent data available as input to the respective algorithms. Many of the tracking algorithms have as a parameter a signal-to-noise (SNR) data-input threshold. Such a threshold may limit the amount of data which is ingested by the algorithm. This will change the data input receiver-operating-characteristic (ROC) point. As thresholds are raised, fewer false alarms and target detections are available to the tracker. Raising this threshold may improve the algorithm performance with regard to false tracks, but it may also degrade the performance in providing good true track holding. In fact, a complete algorithm performance characterization may be obtained by running the tracker multiple times for different input thresholds. The tracker input metrics (PD and FAR) infer an algorithm's SNR threshold setting, and are important in identifying the input ROC operating point. These metrics can then be compared with the tracker output, using the output tracker metrics.

In order to cross-compare different algorithms, it is important that the same input ROC operating point is chosen. Two algorithms without equal access to the same data are sure to yield different results, and the results cannot be quantitatively compared. Therefore, for algorithm cross-comparison, in addition to the use of the common metrics listed above, the same input ROC operating point should be used. The use of the same threshold will ensure this, and the input tracker metrics will reflect this by having equivalent values.

It should be noted that the results presented in this paper may not be entirely conclusive. In some cases, the algorithms have been applied in an optimum fashion and excellent results were obtained. Given that the simulated scenarios are well described and the truth reconstruction is knowable, tracker parameters can be set in a way that may produce better results than if this knowledge were unavailable. Some algorithms are still in development, and results should be considered interim, or preliminary. In other cases, an algorithm may not have been optimized for the particular scenario. As a result, performance in these cases could be considered suboptimum. Future MSTWG performance evaluation efforts will consider "blind" data sets, where complete knowledge of the target and scenario is limited, or even withheld.

3 Tracker Results and Evaluation for the NURC Simulated Scenario

Figure 2 shows the geometry for the NURC simulated scenario [3]. Three ships (red, blue, and green) are heading east, in-line, with an inter-ship spacing of approximately 13 km at 5kts. The target trajectory is shown to the north of the assets (in black) and heading west at 4kts. There are four multistatic nodes (source/receiver pairs) consisting of:

- Node 1: Source (ship 1) – Receiver (ship 1)
- Node 2: Source (ship 1) – Receiver (ship 2)
- Node 3: Source (ship 3) – Receiver (ship 1)
- Node 4: Source (ship 3) – Receiver (ship 2)

Note: there is no source on ship 2 and no receiver on ship 3.

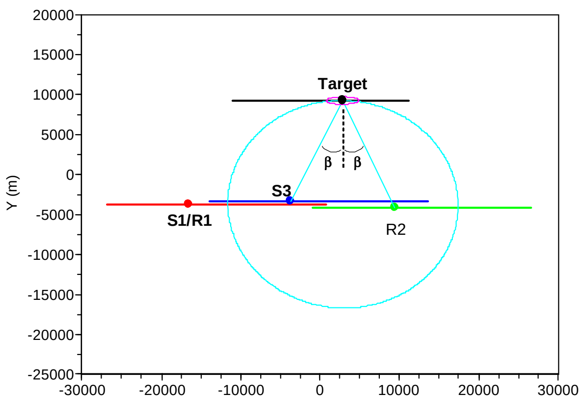

Figure 2. The NURC simulated multistatic sonar scenario (Source 1 – Red; Receiver 1 – Red; Source 3 – Blue; Receiver 2 – Green; Target – Black).

Also shown is a bistatic equi-time ellipse for the target location about 1 hour into the scenario. The run scenario duration is 180 minutes, with both sources transmitting 1-second FM waveforms every 60 seconds.

Available tracker results are shown in the ROC plot shown in figure 3. Each line represents one run of a particular MSTWG tracker on the NURC scenario. A line connects two operating points: the right point is the tracker input operating point and the left point is the tracker output operating point. Each algorithm is shown in a different color. There are four different ROC input operating points (corresponding to different data input thresholds) which have at least two different trackers applied. Each of these four cases is suitable for cross comparison, because they have access to the same input data for tracking. It is seen that only slight changes in threshold yield dramatically different input operating points, due to the false alarm simulations for this scenario.

The results for each of these four cases are summarized in tables 2-5. In general, better performance is obtained when high T-PD and low T-FAR are achieved. For the various cases, T-PD ranges from 0.58 to 1.0 and T-FAR from 0 to 261 false tracks per hour, depending on algorithm type and threshold settings. It is more difficult to infer relative performance for cases which used different input ROC points (i.e., comparing results from different tables).

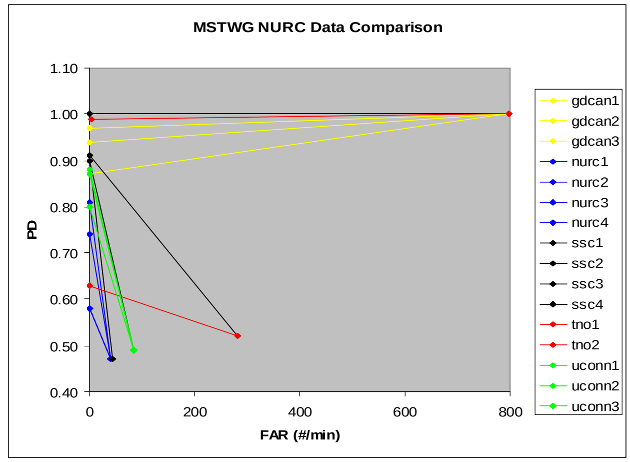

Figure 3. The NURC scenario input/output ROC.

Table 2. Results for full data ingestion.

Input Metrics PD = 1 FAR= 796/min Threshold= -∞ dB	Output T-PD	Output T-FAR (per hr)
GDCAN 1 (distributed, tree depth 2 scans)	0.97	0
GDCAN 2 (centralized, tree depth 2 scans)	0.87	0
GDCAN 3 (centralized, tree depth 3 scans)	0.94	7
SSC-SD 1	1.00	0
TNO 1	0.99	261

Table 3. Results for 13 dB threshold.

Input Metrics PD = 0.52 FAR= 283/min	Output T-PD	Output T-FAR (per hr)
SSC-SD 4	0.91	0
TNO 2	0.67	6

Table 4. Results for 13.5 dB threshold.

Input Metrics PD = 0.49 FAR= 84/min	Output T-PD	Output T-FAR (per hr)
SSC-SD 3	0.90	0
Uconn 1 (ML-PDA, use of amplitude info)	0.87	0
Uconn 2 (ML-PDA, no use of amplitude info)	0.80	1
Uconn 3 (GM-CPHD)	0.88	1.34

Table 5. Results for 13.75 dB threshold.

Input Metrics PD = 0.47 FAR= 42/min	Output T-PD	Output T-FAR (per hr)
NURC 1 (M/N=3/3)	0.81	53
NURC 2 (M/N=4/4)	0.74	8
NURC 3 (M/N=5/5)	0.58	1
NURC 4 (M/N=6/6)	0.58	0
SSC-SD 2	0.90	0

Figure 4 shows the tracker output localization error obtained on the target track for all the applications of all the algorithms. The input (contact) localization error, averaged over all data scans is 682 meters. The results show that in most cases, the output track localization error is smaller than the input, due to the filtering function of the trackers.

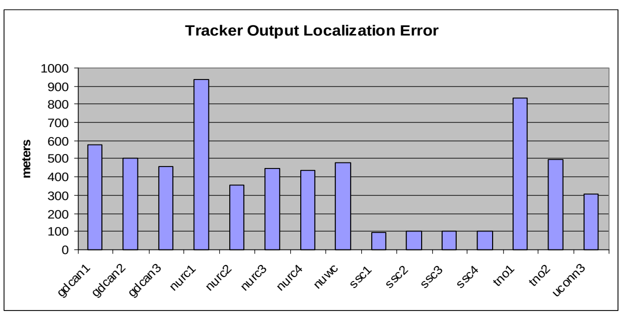

Figure 4. Tracker output localization error.

Figure 5 shows the tracker fragmentation rate for all the applications of all the algorithms. A single track corresponding to the true target covering the entire scenario duration produces a tracker fragmentation rate of 0.33 track segments/hour. Results range from 0.33 to 3.4 segments/hour. Here, all trackers perform relatively well with respect to this metric. A proposal for revising this metric is given is section 5.2.

Figure 6 shows the results for the tracker detection latency metric. This corresponds to the elapsed time from the start of the scenario to the first output of a true confirmed track. In tactical scenarios, one would like the latency to be small, so that prosecution or other action may be taken in a timely manner. In surveillance or area clearance scenarios, this may be less important, because the time frame allows for longer search. The latency values range from 1-2 minutes to slightly over an hour, depending on the algorithms' approach to track initiation.

Figure 7 shows the computer execution rate. This is reported as the fraction of the scenario time (3 hours) needed to process and output results from the tracking algorithm. With one exception, all cases had processing times (on a standard PC) that were faster than the real scenario time.

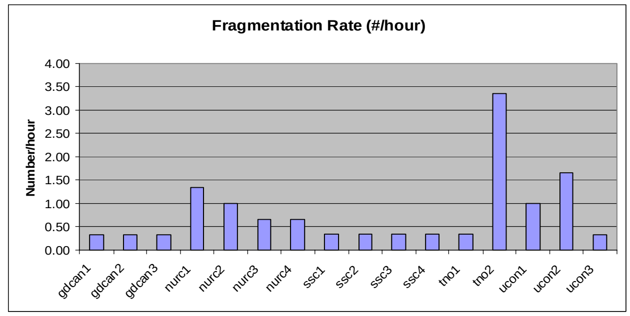

Figure. 5. Tracker fragmentation rate (NURC scenario).

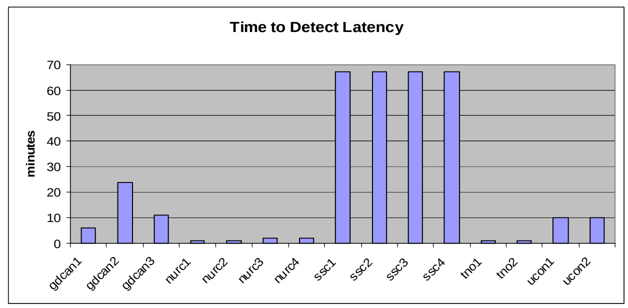

Figure 6. Time to Detect Latency (NURC scenario).

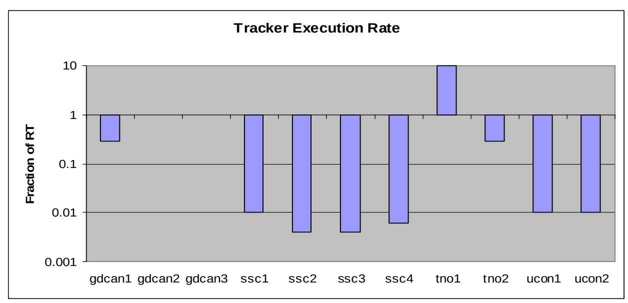

Figure 7. Computer execution rate (as a fraction of real time) for the NURC scenario.

4 Tracker Results and Evaluation for the TNO Simulated Scenario

Figure 8 shows the geometry for the TNO simulated scenario [5]. There are two surface ships, each with a towed sonar source and receiver, providing two monostatic and two bistatic nodes. There are three targets; one is mobile with a "W" shaped trajectory, the other two are fixed (clutter) targets, maliciously inserted near the turns of the mobile target. The scenario is 3 hours in duration and the ping repetition interval is one minute (for both sources). There are a total of 720 scans of data. A summary of the analyses made is given below.

- ARL/UT
  - One tracker run was accomplished.
  - Only the T-LE metric was calculated.
- GDCAN
  - Three tracker runs were accomplished using centralized and distributed architectures. Different thresholds & tree depths were evaluated.
  - The full set of metrics was calculated.
- NURC
  - Two tracker runs were made, with different tracker parameters.
  - There were complications in calculating metrics for this scenario (see discussion in subsequent section).
- TNO
  - One tracker run was made using only the moving target
  - The full set of metrics was calculated.
- UConn
  - ML-PDA: The tracker was run iteratively to get multiple targets.
  - GM-CPHD: one tracker run was
  - Metrics were calculated
  - Only the strongest 10 measurements were used

The GD and TNO cases were run with very close to the same input ROC point (threshold), and therefore can be compared (they have access to about the same data for tracking). Tables 6 and 7 show the tracker output metrics.

Table 6. Output ROC results for TNO scenario.

Input Metrics PD = 0.9 FAR= 384/min (GD) FAR = 434 /min (TNO) Threshold= ~ 13.0 dB	Output T-PD	Output T-FAR (per hour)
GDCAN 1 (distributed, tree depth 2 scans)	0.33	0
GDCAN 2 (centralized, tree depth 4 scans)	0.94	11
TNO	0.92	1

Table 7. Other metrics for TNO scenario.

Input Metrics (same as Table 6)	T-LE	TF	L	ER
GDCAN 1	190	0.0	4	0.15
GDCAN 2	188	7.2	9	0.54
TNO	70	1.7	1	0.66

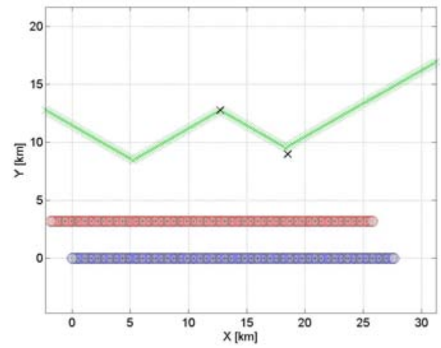

Figure. 8. The TNO simulated scenario. Ships trajectories are shown in red and blue, mobile target track in green, and fixed clutter targets in black.

5 MSTWG Metrics Issues

Application of the trackers to the simulated data has brought to light various issues with the MSTWG performance metrics. This section seeks to clarify these issues, some of which will require further MSTWG discussion and resolution.

5.1 Input PD

The MSTWG definition of input PD assumes no fusion process. The input PD is simply the average of all node PDs of all the available data. This is equivalent to the total data rate available to the tracking algorithm (input data rate). Input PD is a function of detection threshold.

An additional metric used [1], estimates the fusion potential in terms of PD at the tracker input. It is calculated using some fusion rule to determine detectability. For example, on a given source ping, if “one or more” receivers detect, then the system detects. This PD will generally be higher than the one MSTWG uses, and is more related to output PD potential.

5.2 Track Fragmentation

The tracker fragmentation (rate) was previously defined [2] to be

$$TF = \frac{N_{TT}}{T \cdot N_T} \qquad (1)$$

where N_{TT} is the number of true track segments, N_T is the number of true targets, and T is the time duration of the scenario. It was found that using this definition the TF can never reach zero, and further, it will be a function of the scenario duration. An alternate metric definition has been proposed as

$$TF = \frac{N_{TT}}{N_T} \qquad (2)$$

which is more straightforward. This will indicate the number of true track segments (due to fragmentation) that were output over the duration of the scenario, normalized by the number of true targets. Future MSTWG metrics calculations should consider this metric to quantify the negative effect of tracker fragmentation.

5.3 Wandering Tracks

Consider the example tracker output depicted in figure 9. This shows a case where a tracker is effective in holding the target over a portion of the scenario. At a certain point, the track wanders off of the target's true trajectory and becomes false. This may occur in the situation where there is very dense clutter. If the track purity (percentage of target-originated contacts making up the track) is sufficiently high, the track may be designated true rather than false. Alternatively, if the track purity drops low enough (due to the false section), the whole track could be labeled false. The metric calculations in either of these cases will not indicate meaningful values. In the case of the true track which wanders, the T-PD will be overestimated and the LE will be worse than reality.

A potential solution to this problem is to implement a distance (or covariance error) threshold between the tracker position estimate and the true target position. When the threshold is exceeded, the track is manually broken into two pieces, one true and one false. Then the standard metrics may be applied as usual. To report this undesirable performance, a new metric would be reported. This would simply be a count of the number of wander events per target, per scenario.

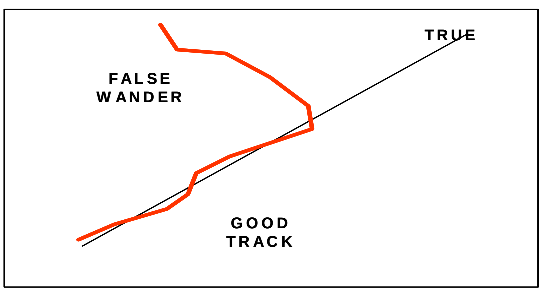

Figure 9. Wandering track example (part true, part false).

5.4 Switching Tracks

Figure 10 shows a example of two targets that cross. The tracker output shows two tracks that erroneously switch assignments. This could also occur when a target passes by a fixed clutter track. Like the wandering track problem, this undesirable behavior will cause problems in the calculation of metrics. However, the same solution may be applied, by splitting the tracks at the point of switched assignment, and then recalculating the standard metrics. To report this undesirable performance, a new metric would be used. This would simply be a count of the number of track switch events per scenario.

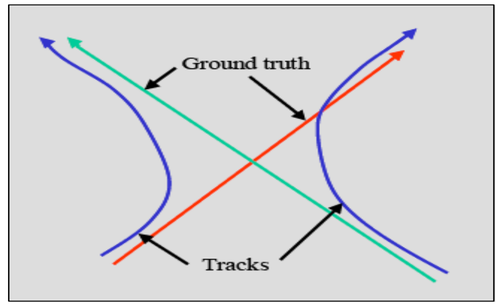

Figure 10. Track switching example.

5.5 Multiple Simultaneous Tracks

Figure 11 shows an example of tracker output where multiple tracks are produced for a single target. This may be the case when signal and information processing produces more than a single contact for the target in the data set. Although information processing should strive to cluster multiple target contacts into one, this may not always be possible. This will complicate the calculation of metrics because there are multiple true tracks to deal with. If this occurs, the effect should be quantified by citing the number of tracks formed on each target and the time duration of the overlapped sections.

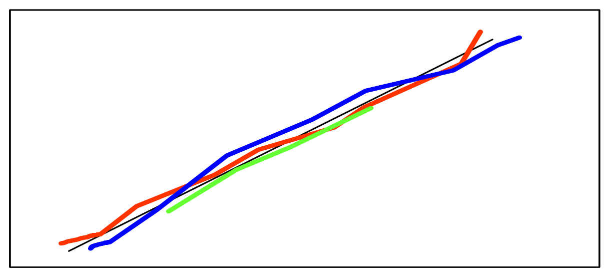

Figure 11. Multiple, simultaneous tracks example (black- target true trajectory, red/green/blue – output tracks)

5.6 Latency

In reviewing the MSTWG results it was evident that a clarification is needed on the latency metric. The tracker detection latency as used in MSTWG is the elapsed time from the beginning of the scenario to the first manifestation of the output of a true target track. This metric therefore will be dependent on the scenario characteristics and the input probability of detection. It only depends on the true target track.

This is not to be confused with the internal tracker processing latency that a tracker may have. This will normally include the time to initialize, confirm, and output a track. For example, in the case of a MHT tracker, it would include the delay due to the hypothesis tree depth. Here the latency will be the same for all output tracks, both true, and false.

5.7 Multiple Targets

The TNO simulated scenario presents the issue of multiple targets in a scenario. This raises the question about how the metrics should be calculated for these cases. This situation will affect the T-PD, LE, TF, and L metrics. Since it may be difficult to assess individual target performance when results of multiple targets are averaged, an alternative to consider is to calculate the relevant target metrics once for each individual target in the data set.

6 Conclusion

The MSTWG is now beginning to yield algorithm comparisons on common data sets with common metrics. The results are preliminary, and additional results will be inserted into this evaluation as they become available. The results are useful in order to understand algorithmic differences; their strengths and weaknesses depending on scenario and performance metrics. This evaluation has highlighted several issues with the MSTWG metrics, which have been addressed. Meaningful comparisons require not only agreement on performance metrics, but agreement to run the various algorithms at the same input threshold (input ROC point), in order to assure that they have the same data available. Future MSTWG tracker analyses conducted on these simulated scenarios should select one or more of the input ROC points already used, to facilitate future comparisons. In general, in appears that better tracker performance is obtained (in terms of T-PD, T-FAR) when longer-duration track initialization schemes are used.

Acknowledgements

This work was sponsored in part by the Office of Naval Research, 321 US. PdT was sponsored by the Royal Netherlands Navy.

References

[1] D. Grimmett and S. Coraluppi, Multistatic Active Sonar System Interoperability, Data Fusion, and Measures of Performance, NURC Technical Report, NURC-FR-2006-004.

[2] S. Coraluppi, D. Grimmett, and P.A.M. de Theije, Benchmark Evaluation of Multistatic Trackers, in Proceedings of the 9th International Conference on Information Fusion, July 2006, Florence, Italy.

[3] D. Grimmett and S. Coraluppi, Contact-Level Multistatic Sonar Data Simulator for Tracker Performance Assessment, in Proceedings of the 9th International Conference on Information Fusion, July 2006, Florence, Italy.

[4] B. La Cour, C. Collins, and J. Landry, Multi-everything Sonar Simulator (MESS), in Proceedings of the 9th International Conference on Information Fusion, July 2006, Florence, Italy.

[5] P.A.M. de Theije and H. Groen, Multistatic Sonar Simulations with SIMONA, in Proceedings of the 9th International Conference on Information Fusion, July 2006, Florence, Italy.

[6] B. La Cour, Bayesian Multistatic Tracking: Results on Simulated Data from the Multistatic Tracking Working Group, in Proceedings of the 9th International Conference on Information Fusion, July 2006, Florence, Italy.

[7] Thomas Lang and Geoff Hayes, Evaluation of an MHT-Enabled Tracker with Simulated Multistatic Sonar Data, in Proceedings of the IEEE Oceans'07 Conference, June 2007, Aberdeen, Scotland.

[8] O. Gerard, S. Coraluppi, C. Carthel, and D. Grimmett, Benchmark Analysis of NURC Multistatic MHT Capability, in Proceedings of the 9th International Conference on Information Fusion, July 2006, Florence, Italy.

[9] S. Coraluppi, C. Carthel, and M. Micheli, DMHT-based Undersea Surveillance: Insights from MSTWG Analysis and Recent Sea-Trial Experimentation, in Proceedings of the 11th International Conference on Information Fusion, July 2008, Cologne, Germany.

[10] C. Hempel, Probabilistic Multi-Hypothesis Tracking for Distributed Multi-Static Active Sonar, in Proceedings of the 9th International Conference on Information Fusion, July 2006, Florence, Italy.

[11] D. Grimmett, Multistatic Target Tracking Using Specular Cue Initiation and Directed Data Retrieval, in Proceedings of the 11th International Conference on Information Fusion, July 2008, Cologne, Germany.

[12] D. Grimmett, Reduction of False Alarm Rate in Distributed Multistatic Sonar Systems through Detection Cueing, Proceedings of the IEEE Oceans'07 Conference, June 2007, Aberdeen, Scotland.

[13] P.A.M. de Theije, L. Kester, and J. Bergmans, Application of the M6T tracker to simulated and experimental multistatic sonar data, in Proceedings of the 9th International Conference on Information Fusion, July 2006, Florence, Italy.

[14] P.A.M. de Theije, in Proceedings of the 11th International Conference on Information Fusion, July 2008, Cologne, Germany.

[15] W. Blanding, P. Willet, and S. Coraluppi, Sequential ML for Multistatic Sonar Tracking, in Proceedings of the IEEE Oceans'07 Conference, June 2007, Aberdeen, Scotland.

[16] O. Erdinc, P. Willet, and S. Coraluppi, The Gaussian Mixture Cardinalized PHD Tracker on MSTWG datasets, in Proceedings of the 11th International Conference on Information Fusion, July 2008, Cologne, Germany.